FEDERATED SIMULATIONS FOR SYSTEMS OF SYSTEMS INTEGRATION

Robert Kewley
James Cook
Niki Goerger
Dale Henderson
Edward Teague

Department of Systems Engineering
United States Military Academy
West Point, NY 10996, USA

ABSTRACT

Systems of systems integration is a difficult engineering challenge that places a particular burden on the engineers who must develop simulation models to support that integration. Developing a large scale stand-alone model to support systems integration is a time-consuming process that is often not possible. An alternative approach is to leverage existing models in a federation. This type of work requires a specialized set of engineering skills. The United States Military Academy Department of Systems Engineering SysHub research program is better defining these skills and applying them to different problem domains. This paper highlights how capabilities for information exchange, environmental representation, entity representation, model development, and data collection support the federation development process.

1 INTRODUCTION

Systems of systems integration is one of the biggest challenges facing military forces today. This challenge exists whether the integration occurs in the acquisition community or in the operational environment. Large acquisition programs such as the Army?s Future Combat Systems (FCS) program are doing parallel development of many systems. Architecture and design changes in one system have ripple effects that propagate across all of the systems in development and to the environment. This creates a complex network of interactions in which architectural changes to one of the subsystems impact other FCS systems, other systems into which FCS must integrate, and soldiers who operate the system. A similar problem exists in the operational community. The evolving wartime environment demands a series of parallel training, acquisition, and fielding initiatives that must be integrated. Analysis of these interactions demands a more robust and reconfigurable simulation paradigm. The United States Military Academy (USMA) Department of Systems Engineering's SysHub program looks to expand the research base and body of knowledge for modeling, simulation, and analysis to support systems of systems integration.

Federated simulations must be as agile, robust, and interactive as the operational environment they support. Large single models that enable analysis of systems of systems are not effective because no single modeling effort can account for all of the complexities. Instead, modeling and analysis must be done in a distributed and parallel fashion, mirroring the development of training, acquisition, and fielding initiatives. This leads to a number of subsystem models that effectively analyze different domains. In order to support systems of systems analysis, subsystem models need to be federated. Current federation architectures such as distributed interactive simulation (DIS) and existing versions of high level architecture (HLA) support interoperability in the military fire and engagement domain. Additional interoperability research is needed to support federations of models that examine other domains, such as command and control and human behavior. In addition, architectural modeling processes and tools that enable agile architecture development and revision will empower federation developers to ensure that interactions between models have shared meaning in both the conceptual and technical domains.

The set of initial projects for SysHub focuses in these research areas. In one project, architectural and integration methods from the software engineering domain are applied to development of a federation to support analysis of soldier systems. In two other projects, federated models are used to investigate the effectiveness of alternative base attack and base defense systems. A final project integrates analysis, gaming, and sensor models to assess the performance of surveillance systems in a complex environment. Synergy occurs as each of these different efforts is a step forward in enabling agile federations to be assembled, run, and analyzed. SysHub's approach does not build models. That is the task of domain experts in their respective development areas. SysHub is a cooperative academic, military, and industry effort that seeks development of the technical and

Report Documentation Page		

1. REPORT DATE **DEC 2008**	2. REPORT TYPE	3. DATES COVERED **00-00-2008 to 00-00-2008**
4. TITLE AND SUBTITLE **Federated Simulations for Systems of Systems Integration**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **U.S. Military Academy,Department of Systems Engineering,West Point,NY,10096**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the Proceedings of the 2008 Winter Simulation Conference, 7-10 Dec, Miami, FL

14. ABSTRACT

Systems of systems integration is a difficult engineering challenge that places a particular burden on the engineers who must develop simulation models to support that integration. Developing a large scale stand-alone model to support systems integration is a time-consuming process that is often not possible. An alternative approach is to leverage existing models in a federation. This type of work requires a specialized set of engineering skills. The United States Military Academy Department of Systems Engineering SysHub research program is better defining these skills and applying them to different problem domains. This paper highlights how capabilities for information exchange environmental representation, entity representation, model development, and data collection support the federation development process.

15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**9**	

conceptual glue, a set of engineering skills, that enables federations of models to support systems of systems research, analysis, and training.

2 LARGE-SCALE MODELS

The Department of Defense has always had great difficulty in developing large-scale models to support analysis. One such program was the Joint Warfare Analysis System (JWARS). This was a multi-year $100 million programming effort designed to develop a single model for the analysis of joint theater campaigns (Borgia 2005). This model attempted to incorporate, among other things, ground, naval, and air warfare, joint mobility, logistics, intelligence, surveillance, and reconnaissance, and command and control. While some level of capability was achieved in all of these areas, the Department of Defense finally stopped funding development of this model after heavy costs and slower than expected development times. At the tactical training level, the Army's OneSAF model has been officially released and is being used in the training community. It has significant capability in the tactical combined arms training arena. However, in its current state, it lacks some desired analysis capabilities in behavior modeling and automated command and control. Similarly, the Army's Training and Doctrine Command has spent several years developing the COMBAT XXI model, and it has just reached initial capability. It is clear from these examples, that the development process for large-scale models is long and difficult. In some cases, an alternative approach is to federate existing models to achieve sufficient capability for the question at hand. The Army's Program Executive Office (PEO) Soldier, in cooperation with the USMA Department of Systems Engineering, has taken this approach to enable analysis of soldier systems. This PEO Soldier Simulation Road Map work is described in section 4.1 (Kramlich 2007). The goal of SysHub is to expedite the federation development process through the identification and development of engineering skills necessary within that process.

3 SYSHUB

SysHub is a capability to rapidly conceptualize, develop, execute, and analyze data using federated simulation models. This concept is illustrated in Figure 1. At the core of this capability is a set of engineering capabilities. The outer ring shows a series of domains in which federated models have proven useful. Within that ring, we list a series of current application areas from those domains. For each application, a different federation of models must be developed to support the question at hand. These federates must be held together, conceptually and technically, by the engineering capabilities in the center of Figure 1. These are the key research areas that enable development of federated models:

- Information Exchange System - The capability to pass meaningful information between federates during the simulation run.
- Environmental Representation - The capability for federates to reference a shared and correlated environment in which entities interact.
- Entity Representation - The capability for federates to referenced shared conceptually aligned information about entities in the simulation. Some of this representation is passed via the information exchange system.
- Models - Within the context of the analysis or training question, the internal models of each federate must be validated and coordinated across the federation.
- Data Collection - The capability to collect meaningful information from the simulation run in the context of the analysis question or training objective for which the federation was designed.

In order for SysHub to work, it must be supported by a systems engineering process for parallel and coordinated development on all of these areas. A good starting point for this process is the IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP) (IEEE Computer Society 2003). However, this process focuses primarily on the federation object model development within information exchange system. It fails to address coordination of the other SysHub research areas. A more comprehensive systems engineering process has been proposed in (Tolk, Litwin, and Kewley 2008). It calls for a common definition of requirements via the Military Missions and Means framework (Sheehan, Dietz, Bray, Harris, and Wong 2004). It then goes on to call for further process of information exchange system development and analysis support using Model Driven Architectures (Object Management Group 2007). The SysHub research program will evaluate and extend these methodologies to better support engineering solutions to environmental representation, entity representation, and modeling challenges.

3.1 Information Exchange System

With respect to software systems, interoperability is "the ability of two or more systems or components to exchange information and to use the information that has been exchanged (Institute of Electrical and Electronics Engineers 1990)." Data standards are specific agreements between agents responsible for different software subsystems that communicate with each other when functioning as parts of a larger system.

An information exchange system provides for a basic level of interoperability between simulations. It is a necessary condition for any higher level of interoperability. For
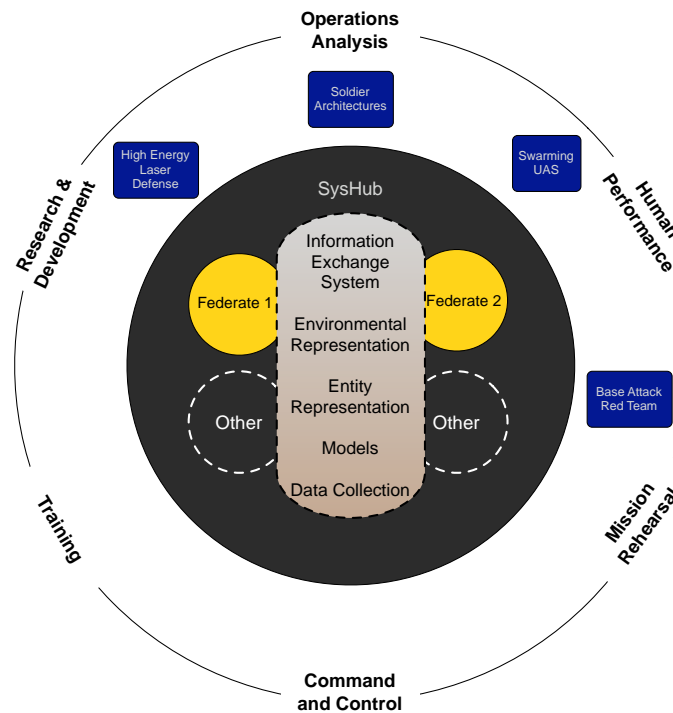
Figure 1: SysHub conceptual model

example two different simulations can not communicate meaningfully about some aspect of the environment if they cannot communicate at all in the first place.

Two common models for providing interoperability are the Distributed Interactive Simulation (DIS) protocol (IEEE-SA Standards Board 1998) and the High Level Architecture (HLA) (IEEE-SA Standards Board 2000). These are established Institute for Electrical and Electronics Engineers (IEEE) standards for allowing simulations to share information at run time. An example of a data standard is the defined federation object model implemented to enable a specific federation instance over HLA. Unfortunately the techniques are generally brittle in one case (DIS) and challenging to implement in the other (HLA), and in both cases the tools built to perform the underlying work (e.g. communication, parsing,) within a federation end up being in practice custom implementations built for a specific purpose. While these implementations may be re-used, the cost of re-use approaches the cost of developing an entirely new implementation. Many excellent examples of these custom instantiation have been developed at significant cost and over long development periods. Many more projects that could have benefited from the use of simulations in federation remain incomplete or unexplored because the challenges of actual implementation are too steep to warrant the cost. Meanwhile, within the greater community, web services

are more and more commonly used to link very different business models, generated for different purposes and at different times by different development teams. These systems are able to execute critical business functions with high reliability using service oriented architectures. Again in practice web enabled service oriented architecture requires considerable investment, but that investment is more likely to have some hope for re-use if it proceeds from some interoperability goals.

One aspect of the SysHub project is exploration of the application of robust, commercial interoperability models, like those used to create service oriented architectures using web services for data transport. This research may be viewed as early implementation test of those revisions to IEEE 1516 aimed at making web services and other standards more common to the commercial world a part of the HLA (Möller and Löf 2007). However the focus of SysHub is on exploring what is possible with very low cost, low overhead communication schemes that use less experiment-specific interoperability standards.

3.2 Environmental Representation

In many simulation abstractions, entities interact with each other and with their environment. In federated simulations, it is not common for the models to share a single representation

of the environment across the network. In most cases, each individual model has its own internal representation of that environment. If the results of the simulation are going to be valid, those environmental representations must be sufficiently correlated so that the results of the simulation are not obscured by differences in these representations.

Environment encompasses natural and manmade physical elements of the battlespace occurring in the terrain, atmosphere, ocean, and space domains. In traditional modeling and simulation taxonomies, buildings and land-based infrastructure fall under the domain of terrain. Not only are features considered more persistent over time included but those things such as weather and obscurants that change over much smaller time scales are also important elements of environment that must be captured in its representation.

In federating or linking in some way different applications, it is key that the environment is correlated for meaningful interchange and execution of a scenario. This is a particularly challenging area based on the fact that there often are different means of defining the relief (elevation over spatial extents of the area of interest) and extracting and representing features. Moreover there are often differing terrain data models underlying applications.

Terrain databases created for different applications are generally not created from the same source data. The source data may be of different resolutions and the techniques for sampling and integrating data may be different. Thus, it is not surprising that roads, bridges, rivers, forested areas, etc., that are the map for different applications do not nicely overlay each other. Additionally, if the surface relief is represented differently across applications, adjudication of line-of-sight and other interactions with the terrain become problematic. These can cause major issues when federating simulations. There is a need for entities in the simulation to have the same notion of where elements in the environment are and to execute off a correlated or the same terrain representation.

The terrain data models include features such as road segment or surface element. These will have further attribution that denotes things such as type of road or soil type. Furthermore, there are allowable enumerations for road type and for soil type. Consider, soil type for example. Enumerations might be those denoted by the U.S. Geological Survey (USGS) soil classifications or they might be more simply represented as sand, clay, mud, or gravel. If linking applications using these two different soil type enumerations and considering ground vehicle mobility, it is important to understand how this affects the underlying algorithms computing mobility and how this affects a fair fight, interoperability, and correlation. There have been various efforts to standardize data to include the Standards efforts Synthetic Environment Data Representation and Interchange Specificaiton (SEDRIS) which is an open project with several products and participates in standards development (SEDRIS 2007).

The correlated generation of terrain information for simulation databases is a very broad and complex topic. The SysHub research program will investigate the relative effectiveness of three options with respect to terrain representation. The first option is to have each model interact with terrain using copies of the same terrain data model and associated programming interface for terrain algorithms such as mobility factors or line of sight. A second option is to develop correlated data from the same geospatial data sources in a single tool. There are several commercial and government-owned tools that use a common set of raw geospatial feature data, elevation data, and imagery to generate simulation databases in multiple formats. The final option is to independently generate correlated databases using different tools. This option may introduce correlation errors due to differences between the tools. It is possible to manually correlate the resulting databases using common reference points, but this is a difficult and imprecise process. Regardless of the option used, it is important to validate the effects of terrain data on the scenario during the federation test phase. Checking the validity of observed movement rates, line of sight algorithms, and visual representations will give confidence to the accuracy and correlation of environments across the federated models.

SysHub presents a significant opportunity for deliberate experimentation focused on examining issues pertaining to correlated environment representation. Examining the effects of differing resolution, differing representations, and miscorrelations and the effects on the outcomes of experiments is needed for the community. There are opportunities to develop methodologies to examine these higher order effects that are often accepted without analyzing the ramifications. In doing so, bounds on what applications to link and what issues to address readily versus those that will require some additional up front work can be generated.

3.3 Entity Representation

In order to construct a valid federation, member federates must achieve conceptual and data alignment for the entities represented. Their attributes, states, development, domain, and range are functions of the federations construct. Attributes represent an entitys characteristics with data values. As a model grows more sophisticated its entities often exhibit more attributes and states. Each entity has its own history. A federate keeps a record for its entities, tracking their attributes and state. The nature of the federation dictates the location and composition of that record. For instance, entities may be split up amongst the federates. However, its not valid to assume that a consolidated record of entities and their states exists unless it is developed as a part of the larger design.

Typically, "state" refers to the state of the simulation. Averill Law refers to state as a "collection of variables
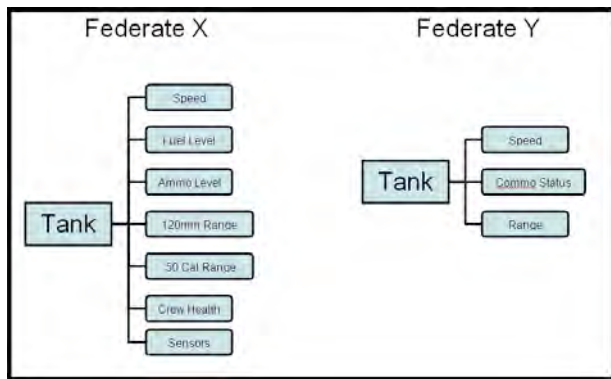
Figure 2: Entity attribute alignment

necessary to describe a system at a particular time, relative to the objectives of a study (Law 2007)." In this light, the state of an entity is the collective state of each of its attributes. Given the environment, e.g. interaction with other entities or changes in other entities that influence it, changed attributes represent changed states. The challenge within the federation is timely and consistent application of each entitys attributes and their states throughout.

As an outcome of the system design, operational definitions and attributes must be vetted and aligned in detail for verification and validation. Assume that entity A is a tank. The federates recognize it as a highly survivable vehicle that moves throughout the environment, communicates with others, and conducts direct fire engagements. However, the tanks attributes may vary in each federate. Attributes like azimuth, speed, mobility, survivability, ranges, onboard weapons, accuracy, ammunition supply, will likely be inconsistent. Since the federates do not each regard a tank using the same attributes, interactions can be flawed. For instance, one federate may characterize the tank as an M1A1 while another may use an M48. Even if federates manage to share a common tank type, it is possible that they each describe the tank with a different number of attributes. Those attributes will not neatly map to one another, Figure 2. This could greatly undermine confidence in results. Federation development requires not only validation of entity representation within each federate, but also validation of the differences between these representations across the federation. The differences should not affect the analysis results to an extent they are no longer useful for the question at hand.

Outcomes from the interactions between federates resulting from direct or indirect contact must be examined for their consistency. Go back to the tank example. If the vehicle rolls over a mine, the federates involved in the interaction and those that record the data must generate an outcome common to all. If its a direct hit? Is it a mobility

kill, complete destruction, or a null event? What is the effect on the crew? Is it necessary to track them as separate entities or are they just a part of the tank? Ensuring consistency within a federation requires patience an intimate knowledge of each federate. Thus, subject matter experts on each federate must approach the simulation development as a process where the environment, its entities, and their attributes are vetted. It is the system design and desired study outcome that will drive the process.

3.4 Models

When composing models in a federation, model validation requires some particular considerations. A shallow application of interoperability might only seek to align models and federates by converting the simulation's data into the necessary input and output format specified by the federates. This can often be done at the programmer level. Support for interoperability protocols such as HLA and DIS often enable new federates to be "plugged in" with little or no programming or engineering. The danger in this type of integration is that the federated model may not be designed to perform analysis in the new context. Just because a model has been validated in one context does not mean that it can automatically be used properly in a new context. Composing valid federations requires some additional steps.

The subject matter experts for the federated model must work with the simulation systems engineers to validate the model in its new context. While model validation is a complex subject beyond the scope of this paper, a few key points aid the discussion. First, it is assumed that the federated model has already been validated in some context. If this is the case, the modeling assumptions from the original validation must be checked against the new simulation context to ensure that this new context does not invalidate these assumptions to a point where the model is not useful in the new context. Once this static validation is complete, further validation can take place during the federation test and evaluation phase. In this phase, data inputs and outputs from the model are checked against know quantities to ensure the model behaves properly in the new context. Finally, a subject matter expert in the domain to be modeled can visualize or check model runs within the new context to give "face validity" to the model in its new context.

3.5 Data Collection

Successful analysis requires a clear definition of the questions to be answered and an understanding of the extent to which the questions can be accurately represented and answered by the tools available to the analyst. There are two primary tasks involved in moving from questions to answers. First, the question must be translated to a form

representable in the simulation, and criteria must be established for translating the results of the simulation back to answers to the original questions. Second, the simulation must be run and enough data collected to answer the question to the desired level of confidence.

In order to translate the questions as accurately as possible to a form answerable by simulation, the analyst must understand the modeling assumptions made by the designers of the simulation. A completely translated question must be in a form directly answerable by simulation. It should be phrased in terms of simulation inputs and outputs only, and a translation must be defined from its answers to answers to the original questions.

Once translated, the question establishes a requirement for data collection. If the tools available to the analyst do not support collection of the data required, tools must be adapted or developed to meet the need or the scope of the question must be reduced to fit the available data.

Current tools for data collection tend to fall into two broad categories. The first consists of standalone loggers which record raw data from DIS, HLA, or other data interchange systems. The other consists of integrated data collection and analysis systems that record information at a level of abstraction comparable to a simulation's internal object model. Integrated data collection systems are easier to use as long as they provide the data required. When other data is required, it is often possible to construct small ad-hoc tools to analyze and summarize the output of the lower-level loggers.

In addition to automated tools, human observation is a valuable tool for data collection. In some cases, the most practical way of capturing data may be to have a human expert watch or participate in events as they unfold. In others, human psychological or physiological performance may be an important part of the experiment, and must be measured. Thus visualization tools and immersive environments are also an important part of the analyst's data-collection toolset.

4 APPLICATIONS

The USMA Department of Systems Engineering is working on several applications for which the SysHub concept helps guide federation development and supporting analysis. In each case, unique challenges from the problem domain call for slightly different application of SysHub's engineering skills.

4.1 PEO Soldier Simulation Road Map

The Army's Program Executive Office (PEO) - Soldier has the complex task of acquiring and integrating a system of soldier equipment that meets their mission requirements. In order to better assess trade-offs in different soldier architectures, they seek an improved simulation capability that better

represents the individual soldier on the battlefield. No single model provides this capability. They are pursuing a strategy of integrating three different simulation models to take advantage of the strengths of each. These models are the Infantry Warrior Simulation (IWARS), One Semi-Automated Forces (OneSAF), and the Combined-Arms Analysis Tool for the 21st Century (COMBAT^{XXI}). The engineering skills represented by SysHub have supported this integration in several important ways.

With respect to an information exchange system, the team is using a form of the HLA called the Modeling Architecture for Technology, Research, and Experimentation (MATREX) (Hurt, McKelvey, and McDonnell 2006). Two of the three component models already had developed support for this architecture. In order to ensure common environmental representation, each modeling team incorporated the OneSAF Environmental Runtime Component. This ensured common terrain representation with respect to both data and algorithms. As the architecture matured, common entity representation became a significant challenge. Entities such as vehicles and munitions had to be represented in a conceptually aligned way with common performance data. Since each model had its own mature set of data representations, significant work had to be done to ensure common naming and performance data. With respect to data collection, the team is employing a parallel strategy of collecting some data from the federation using MATREX tools and collecting other data from the internal data collection tools of the federates. The analysis questions will dictate which is more useful.

4.2 Swarming Unmanned Aircraft Systems

A second research application is the development of a federation will support research of self-organizing semi-autonomous Unmanned Aerial Vehicles (UAVs). The planned federation will also require human in the loop (HITL) inputs, dynamic artificial intelligence algorithms, and flexibility to compare, limit, and experimentally vary both. We will program a wide range of behaviors, both parametrically and constructively.

The federates are specifically chosen to meet the needs of the experimental question. Specifically, we are interested in how to assign various rule sets that generate semi-autonomous behavior in UAVs. The UAVs will perform their tasks in a relatively busy operational environment, including civilians, urban terrain, and friendly and enemy forces. The UAVs will search for a variety of specific targets. We will collect data that characterizes the success or failure of the UAVs as they execute assigned tasks. The quality of the outcome of the search will drive selection of better self-organizing algorithms.

The effort brings together a collection of simulations performing different roles. MÄK's VR Forces simulation
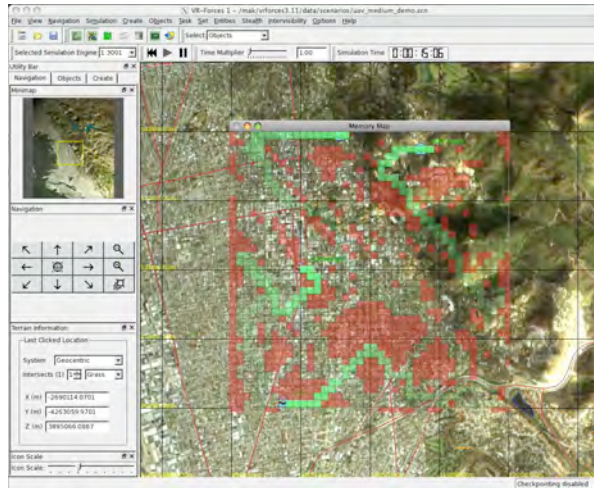
Figure 3: Autonomous UAV modeling with a visualization federate overlaid.

was used to provide basic environmental and entity capabilities. A specific behavior federate was written and integrated with VR Forces to capture the ability to test different rule sets and a pheromone memory map. A visualization federate was written to overlay VR Forces and show the memory map correlated with the simulation picture as seen in Figure 3. This federate, along with a data collection federate, used DIS as the information exchange system.

Future expansion of this project into virtual experiments will require the addition of visual federates that display UAS video. This federate will bring new challenges with respect to entity representation and models. Specifically, the sensor video displayed in the visual sensors must match the capabilities of the sensors modeled in VR Forces. The internal data structures and modeling of these sensors must by synchronized against a common model-independent set of capabilities.

Data collection is a combination of ad hoc methods and the MAK DIS Logger. The logger captures PDUs as they are broadcast from all federates. This provides a record of the events and a rapid and easy playback of federation exercise iterations. However the session log for each iteration can become unwieldy rather quickly. For some specific data, we also use a real time scoreboard and other tools to enhance the federations analysis capability.

4.3 High Energy Laser Base Defense

Another application example comes from research into the use of high energy laser systems on the battlefield. High energy laser capability has grown dramatically, and these devices show promise across a wide range of engagement scenarios, from theater ballistic missile defense to counter sniper systems. This particular example considers a high

energy laser as a base defense system for shooting down incoming mortar projectiles. Laser performance models, such as HELIOS-CR are high fidelity physics models which provide information about beam quality, power, and thermal effects on the target (MacFarlane, Golovkin, and Woodruff 2007). They require detailed information about, for example, atmospheric conditions that conventional combat simulations typically ignore. We would not want to ignore these details in a simulation designed to test the effectiveness of a laser system. However, models like HELIOS tell us nothing about the engagement process, nor do they provide information about the impact of placing the laser system in a general combat scenario, battle or campaign. In order to model engagement process (from detection through engagement and assessment) we need something like the Extended Air Defense Simulation (US Army Space and Missile Defense Command 2008). EADSIM can then

> "easily be federated with campaign-level models for analysis, training, and exercises; with other mission-level models or functional area models such as artillery and land combat models; with high-fidelity models for specialty functions such as detailed air to air combat; and with virtual simulators, allowing a live gunner or pilot to operate a training simulator that interacts with the full scenario. In these federations, EADSIM provides a variety of functions, such as providing real-time computer generated forces, playing Tactical Ballistic Missile warfare, or commanding aircraft transferred to other simulations." (US Army Space and Missile Defense Command 2008)

While this application is still in development, a couple important lessons can be learned about federated modeling. With respect to environmental modeling, the aspect of weather becomes important because atmospheric conditions effect laser beam dispersions. An even more subtle lesson can be derived with respect to models and model validation. In this case, the state of the target projectile can be derived from the EADSIM campaign-level model. This allows computation of distance, angle of attack, and projectile velocity in the HELIOS engagement model. A simple approach to interfacing these models would seem to provide the HELOIS engagement model with all of the necessary data to compute whether the laser destroyed the projectile. However, a domain expert could point out that the HELIOS model assumes a static target, but the mortar round is spinning in its flight. This dissipates the laser energy from a single point to a ring around the circumference of the projectile, and much more energy is needed to defeat

it. This example highlights the importance of validating a federated set of models using domain experts who understand the underlying assumptions of each model, instead of naively transferring input-output data from one model to the next.

5 Conclusions

From the application areas in Department of Systems Engineering research program, it is clear that continued development of the SysHub concept will enable the use of federated models for research, analysis, human integration, command and control, training, and rehearsal systems. At the core of this effort, is the development of engineering skills to support federating existing models to solve challenging problems. The department is in the early stages of work that will better define and specify these skills for future efforts. Additional work is also ongoing to build an detailed engineering process, with tool support, that will give a better road map to systems engineers engaged in federation development.

REFERENCES

Borgia, M. 2005. JWARS enters a new phase. *Phalanx* 38 (4): 3–6.

Hurt, T., T. McKelvey, and J. McDonnell. 2006. The modeling architecture for technology, research, and experimentation. In *Proceedings of the 2006 Winter Simulation Conference*, ed. L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, 1261–1265.

IEEE Computer Society. 2003. IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP). Technical Report 1516.3.

IEEE-SA Standards Board. 1998. Standard for distributed interactive simulation - application protocols. Technical Report IEEE 1278.1A-1998.

IEEE-SA Standards Board. 2000, September. IEEE standard for modeling and simulation (M&S) high level architecture (HLA) - framework and rules. Technical Report IEEE 1516-2000.

Institute of Electrical and Electronics Engineers. 1990. IEEE standard computer dictionary: A compilation of IEEE standard computer glossaries. New York.

Kramlich, G. 2007. PEO Soldier simulation roadmap III: Initial working federation. Technical Report DSE-TR-0704, United States Military Academy Operations Research Center, West Point, NY.

Law, A. M. 2007. *Simulation, modeling, and analysis*. 4th ed. New York: McGraw-Hill.

MacFarlane, J. J., I. E. Golovkin, and P. R. Woodruff. 2007. *HELIOS - A 1-D radiation-magnetohydrodynamics code with inline atomic kinetics modeling*. PRISM Computational Sciences.

Möller, B., and S. Löf. 2007. A management overview of the HLA evolved web service API. In *Proceedings of the 2007 Fall Interoperability Workshop*. Simulation Interoperability Standards Organization.

Object Management Group. 2007. OMG model driven architecture. http://www.omg.org/mda. accessed 14 march 2008.

SEDRIS. 2007. SEDRIS technologies. http://www.sedris.org. accessed 14 april 2008.

Sheehan, J. H., P. H. Dietz, B. E. Bray, B. A. Harris, and A. B. Wong. 2004. The military missions and means framework. Technical Report TR-756, Army Material Systems Analysis Activity.

Tolk, A., T. Litwin, and R. Kewley. 2008, December. A systems engineering process for driving federated model development with operational requirements. In *Draft submitted to Proceedings of the 2008 Winter Simulation Conference*, ed. S. J. Mason, R. R. Hill, L. Moench., and O. Rose.

US Army Space and Missile Defense Command. 2008. *EADSIM executive summary*.

AUTHOR BIOGRAPHIES

ROBERT H. KEWLEY is currently the director of the Operations Research Center in the USMA Department of Systems Engineering. He was commissioned in 1988 from the United States Military Academy as an armor officer. His analysis experience includes a teaching assignment at West Point and a tour at the Center for Army Analysis. LTC Kewleys research interests focus on command and control systems. He has a Master of Science in Industrial Engineering and a Ph.D. in Decision Science and Engineering Systems, both from Rensselaer Polytechnic Institute. His email address for these proceedings is `<Robert.Kewley@usma.edu>`.

DALE L. HENDERSON is an assistant professor in the USMA Department of Systems Engineering. LTC Hendersons research interests include non-linear optimization and discrete event simulation. He has a Master of Science in Operations Research from the Naval Postgraduate School and a Ph.D. in Systems and Industrial Engineering from The University of Arizona.

NIKI C. GOERGER is a research engineer with the U.S. Army Engineer Research and Development Center (ERDC). Her expertise is in the area of physics-based and effects-based representation and quantitative analysis in modeling and simulation (M&S) for military applications. She is currently a research associate at the United States Military Academy with research tracks in analysis for

countering improvised explosive devices, M&S and Battle Command interoperability, and physics-based representation in simulations. She received her BS in Biological Engineering and MS in Agricultural Engineering from Mississippi State University and her Ph.D. in Industrial Engineering from Texas A&M University.

EDWARD TEAGUE is a currently serving as an analyst in the Operations Research Center of Excellence and an Instructor in the Department of Systems Engineering at the United States Military Academy, West Point, New York. He earned his Bachelor of Science from the United States Military Academy in 1995 and his Masters of Science in 2006 from the University of Texas. MAJ Teague has served as an aviator with the 25th Infantry Division, 10th Infantry Division, and 8th Army. His research interests include simulation, system of systems analysis, and reliability.